

What is claimed is:

1. A method comprising:
 - receiving a plurality of non-native instructions in a selected one of a source form and an intermediate form; and
 - compiling the plurality of non-native instructions to generate object code for the non-native instructions, including optimizing the generated object code to improve at least a selected one of a power level required and an amount of energy required to execute the generated object code in a target execution environment.
2. The method of claim 1, wherein said receiving comprises receiving the non-native instructions in a byte code form.
3. The method of claim 1, wherein said optimizing comprises analyzing an object code segment for execution power level requirement, and determining whether an alternative object code segment with lower execution power level requirement is available.
4. The method of claim 1, wherein said optimizing comprises analyzing an object code segment for execution energy consumption, and determining whether an alternative object code segment with lower execution energy consumption is available.
5. The method of claim 1, wherein
 - the method further comprises executing the non-native instructions for an initial number of times; and
 - performing said compiling only after executing the non-native instructions for said initial number of times.
6. The method of claim 5, wherein the method further comprises determining the initial number of times the received non-native instructions are to be executed before performing compiling the received non-native instructions.

7. The method of claim 6, wherein the method further comprises
- monitoring said compiling for power level required to perform compilation; and
 - updating a current understanding of power level required for compilation used to determine the initial number of times received non-native instructions are to be executed before compiling the received non-native instructions, if said monitoring observes a power level required for compilation to be different from the current understanding.
8. The method of claim 6, wherein the method further comprises
- monitoring said compiling for amount of energy required to perform an average compilation; and
 - updating a current understanding of amount of energy required for an average compilation used to determine the initial number of times received non-native instructions are to be executed before compiling the received non-native instructions, if said monitoring observes an amount of energy required for compilation to be different from the current understanding.
9. The method of claim 1, wherein the generated object code comprises a plurality of native instructions, and the method further comprises
- monitoring execution of the generated object code for power level required to execute the native instructions; and
 - updating power level requirements of selected ones of the native instructions if said monitoring observes power level requirements for the selected ones of the native instructions to be different from current understandings of the power level requirements of the selected ones of the native instructions.
10. The method of claim 1, wherein the generated object code comprises a plurality of native instructions, and the method further comprises
- monitoring execution of the generated object code for amount of energy required to execute the native instructions; and
 - updating energy requirements of selected ones of the native instructions if said monitoring observes energy requirements for the selected ones of the native instructions

to be different from current understandings of the energy requirements of the selected ones of the native instructions.

11. In an electronic device, a method of operation, comprising:

- receiving a plurality of non-native instructions;
- executing the non-native instructions for an initial number of times; and
- compiling the non-native instructions into object code after executing the received

non-native instructions for said initial number of times.

12. The method of claim 11, wherein the method further comprises determining the initial number of times the received non-native instructions are to be executed before performing compiling the received non-native instructions.

13. The method of claim 11, wherein the method further comprises

- monitoring said compiling for a compilation requirement employed in determining the initial number of times received non-native instructions are to be executed before compiling; and
- updating a current understanding of the compilation requirement if said monitoring observes the compilation requirement to be different from the current understanding.

14. The method of claim 11, wherein the generated object code comprises a plurality of native instructions, and the method further comprises

- monitoring execution of the generated object code for execution requirements of the native instructions; and
- updating execution requirements of selected ones of the native instructions if said monitoring observes execution requirements for the selected ones of the native instructions to be different from current understandings of the execution requirements of the selected ones of the native instructions.

15. An article of manufacture comprising:

- a computer readable medium; and

- a plurality of instructions designed to implement a compiler to compile non-native instructions to generate object code for the non-native instructions, including optimizing the generated object code to improve at least a selected one of a power level required and an energy required to execute the generated object code.

16. The article of claim 15, wherein said optimizing comprises analyzing an object code segment for execution power level requirement, and determining whether an alternative object code segment with lower execution power level requirement is available.

17. The article of claim 15, wherein said optimizing comprises analyzing an object code segment for execution energy consumption, and determining whether an alternative object code segment with lower execution energy consumption is available.

18. An article of manufacture comprising:

- a computer readable medium; and
- a plurality of instructions designed to implement a runtime manager equipped to receive a plurality of non-native instructions, execute the non-native instructions for an initial number of times, and invoke a compiler to compile the non-native instructions into object code after executing the received non-native instructions for said initial number of times.

19. The article of claim 18, wherein the runtime manager is further equipped to determine the initial number of times the received non-native instructions are to be executed before performing compiling the received non-native instructions.

20. The article of claim 18, wherein the runtime manager is further equipped to

- monitor said compiling for a compilation requirement employed in determining the initial number of times received non-native instructions are to be executed before compiling; and
- update a current understanding of the compilation requirement if said monitoring observes the compilation requirement to be different from the current understanding.

21. The article of claim 18, wherein the generated object code comprises a plurality of native instructions, and the runtime manager is further equipped to

- monitor execution of the generated object code for execution requirements of the native instructions; and
- update execution requirements of selected ones of the native instructions if said monitoring observes execution requirements for the selected ones of the native instructions to be different from current understandings of the execution requirements of the selected ones of the native instructions.

22. A system, comprising:

- a storage medium having stored therein a plurality of instructions implementing a compiler to compile non-native instructions to generate object code for the non-native instructions, including optimizing the generated object code to improve at least a selected one of a power level required and an energy required to execute the generated object code; and
- a processor coupled to the storage medium to execute the instructions implementing the compiler.

23. The system of claim 22, wherein said optimizing comprises analyzing an object code segment for execution power level requirement, and determining whether an alternative object code segment with lower execution power level requirement is available.

24. The system of claim 22, wherein said optimizing comprises analyzing an object code segment for execution energy consumption, and determining whether an alternative object code segment with lower execution energy consumption is available.

25. The system of claim 22, wherein the apparatus further comprises a wireless communication interface to receive the non-native instructions.

26. A system, comprising:

- a communication interface to receive a plurality of non-native instructions;
- a storage medium coupled to the communication interface, and having stored therein a plurality of instructions designed to implement a runtime manager equipped to execute the received non-native instructions for an initial number of times, and invoke a compiler to compile the non-native instructions into object code after executing the received non-native instructions for said initial number of times; and
- a processor coupled to the storage medium to execute the instructions implementing the runtime manager.

27. The system of claim 26, wherein the runtime manager is further equipped to determine the initial number of times the received non-native instructions are to be executed before invoking the compiler to compile the received non-native instructions.

28. The system of claim 26, wherein the runtime manager is further equipped to

- monitor said compiling for a compilation requirement employed in determining the initial number of times received non-native instructions are to be executed before compiling; and
- update a current understanding of the compilation requirement if said monitoring observes the compilation requirement to be different from the current understanding.

29. The system of claim 26, wherein the generated object code comprises a plurality of native instructions, and the runtime manager is further equipped to

- monitor execution of the generated object code for execution requirements of the native instructions; and
- update execution requirements of selected ones of the native instructions if said monitoring observes execution requirements for the selected ones of the native instructions to be different from current understandings of the execution requirements of the selected ones of the native instructions.

30. The system of claim 26, wherein the communication interface is a wireless communication interface.